

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

**As rescanning documents *will not* correct images,
please do not report the images to the
Image Problems Mailbox.**

This Page Blank (uspto)

①9 RÉPUBLIQUE FRANÇAISE
INSTITUT NATIONAL
DE LA PROPRIÉTÉ INDUSTRIELLE
PARIS

①1 N° de publication :
(à n'utiliser que pour les
commandes de reproduction)

2 701 578

②1 N° d'enregistrement national : 93 01735

⑤1 Int Cl^s : G 06 F 12/02 , G 11 C 16/00

⑫

DEMANDE DE BREVET D'INVENTION

A1

②2 Date de dépôt : 16.02.93.

③0 Priorité :

④3 Date de la mise à disposition du public de la
demande : 19.08.94 Bulletin 94/33.

⑤6 Liste des documents cités dans le rapport de
recherche préliminaire : *Se reporter à la fin du
présent fascicule.*

⑥0 Références à d'autres documents nationaux
apparentés :

⑦1 Demandeur(s) : *Société Anonyme dite: GEMPLUS
CARD INTERNATIONAL — FR.*

⑦2 Inventeur(s) : Gordons Edouard.

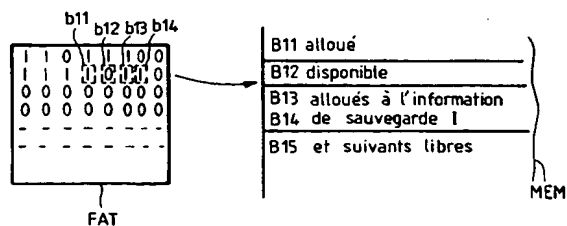
⑦3 Titulaire(s) :

⑦4 Mandataire : Cabinet Ballot-Schmit.

⑤4 Procédé d'écriture dans une mémoire non volatile, notamment dans une carte à mémoire.

⑤7 L'invention concerne les systèmes de gestion de mé-
moire non volatiles (MEM).

Pour éviter des pertes d'informations en cours d'écriture,
on verrouille les séquences critiques d'écriture. Une infor-
mation de sauvegarde (I) est mise en mémoire avant l'exé-
cution de la section critique; le verrou est constitué par les
bits (b13, b14) de la table d'allocation (FAT) qui désignent
l'emplacement (B13, B14) de l'information sauvegardée. Le
verrou est effacé à la fin d'une séquence normale d'écrit-
ture. S'il y a interruption anormale d'écriture pendant la
section critique, le verrou reste verrouillé. Ceci est détecté
à la remise sous tension et une reprise d'écriture est effec-
tuée à l'aide des informations sauvegardées. Selon l'inven-
tion, le verrou et les informations sauvegardées sont dans
une zone variable de la mémoire, ce qui évite la fatigue de
la mémoire en cas d'utilisation intensive. De plus, la ges-
tion de la mémoire est originale en ce sens que deux stra-
tégies d'allocation de mémoire différentes sont utilisées,
pour permettre de détecter par la table d'allocation la pré-
sence d'une information dont on ne connaît pas l'emplace-
ment.



FR 2 701 578 - A1



PROCEDE D'ECRITURE DANS UNE MEMOIRE NON VOLATILE,
NOTAMMENT DANS UNE CARTE A MEMOIRE

L'invention concerne principalement les cartes à mémoire, et tout particulièrement celles qui comportent à la fois une mémoire non-volatile électriquement programmable (EPROM ou EEPROM) et un microprocesseur
5 pour gérer les informations contenues dans la mémoire. Mais l'invention peut s'appliquer plus généralement à tout système de stockage non volatiles de données.

L'invention sera exposée d'abord à propos de cartes à mémoires, et on expliquera ensuite les conséquences
10 qu'on peut en tirer plus généralement pour d'autres systèmes de gestion de mémoire.

Par nature, la carte à mémoire doit pouvoir remplir sa fonction opérationnelle dans des conditions qui peuvent être sévères. Certains composants contiennent
15 des dispositifs permettant de détecter les agressions physiques contre la puce contenue dans la carte. Mais jusqu'à maintenant on n'a pas trouvé de moyen pour protéger complètement la carte dans le cas où elle est arrachée du lecteur pendant une opération d'écriture
20 d'informations dans la carte.

Une manière d'assurer cette protection consiste à faire exécuter par la carte, au moment de la remise sous tension, la reprise de l'écriture interrompue. Cela suppose d'une part qu'on sait détecter qu'il y a eu
25 arrachement, et d'autre part qu'on peut reconstituer les informations qui n'ont pas pu être écrites du fait de l'arrachement.

Les systèmes qui permettent de faire cela sont fondés sur un principe de protection des sections
30 critiques des séquences d'écriture. Le séquençement

d'une opération d'écriture est contrôlé par le microprocesseur, et certaines étapes du programme exécuté par le microprocesseur sont considérées comme des étapes critiques. A chaque entrée dans une section critique, on signale cette entrée en positionnant un "verrou logique" dans un état logique déterminé. Le verrou est un élément de mémoire non volatile dont l'état haut ou bas définit un état verrouillé ou déverrouillé. A la sortie de la section critique du programme, le verrou est remis dans son état initial déverrouillé. C'est la lecture de ce verrou, physiquement figé lorsque la mémoire cesse d'être alimentée, qui permet de savoir, lors de la remise sous tension de la carte, que l'écriture a été anormalement interrompue (par exemple par arrachement de la carte) alors que la section critique du programme était justement en cours d'exécution.

Si on désire procéder ainsi, l'écriture d'une information INF à une adresse A de la mémoire peut alors se dérouler par exemple en deux temps :

- on écrit l'information INF et l'adresse A dans une zone Z de la mémoire non volatile; la zone Z conserve temporairement ces données;

- on écrit définitivement l'information INF à l'adresse A.

La seconde étape est considérée comme une section critique : on place donc le verrou dans l'état "verrouillé" au début de cette étape, puis on le déverrouille à la fin de l'étape si tout s'est déroulé correctement.

Si tout s'est passé correctement, on libère ensuite le contenu de la zone Z qui va servir à nouveau lors d'une opération d'écriture ultérieure.

Si un arrachement s'est produit pendant la section

critique, le verrou se trouve dans l'état verrouillé lorsqu'on remet la carte sous tension; la procédure d'initialisation prévoit obligatoirement d'aller examiner l'état du verrou; s'il est dans l'état verrouillé, on en tire la conséquence qu'il faut exécuter à nouveau la deuxième étape, ce qui est possible puisque l'information INF et l'adresse A ont été conservées en mémoire non volatile dans la zone Z.

La fréquence d'utilisation de la carte peut être très importante. La zone Z de sauvegarde temporaire risque alors d'être utilisée intensivement. L'élément de mémoire qui sert de verrou est aussi utilisé intensivement. Pour les mémoires EEPROM tout particulièrement, une utilisation intensive peut conduire à une dégradation de la fiabilité de l'information mémorisée. Pour diviser ce risque, on peut utiliser alternativement deux zones Z1 et Z2. Le verrou reste cependant constitué toujours par le même élément de mémoire et il subit un nombre important d'utilisations.

L'invention a pour buts :

- d'abord de proposer une solution pour réduire les risques liés à l'utilisation intensive d'une zone de mémoire temporaire pour les cartes à mémoire dans lesquelles le système d'exploitation de la carte prévoit une écriture en deux étapes avec section critique et verrou;

- mais plus généralement aussi d'accroître les possibilités de gestion des mémoires non volatiles, que ce soit pour des cartes à mémoire ou non.

En effet, l'idée développée selon l'invention pour atteindre le premier but a des conséquences qui sont beaucoup plus générales.

Selon un premier aspect, l'invention prévoit que

l'information sauvegardée en vue de l'exécution d'une section critique est placée à un endroit variable de la mémoire, défini par le système d'exploitation en fonction des espaces libres au moment de cette sauvegarde. L'emplacement de stockage de l'information de sauvegarde ne sera donc pas toujours la même zone Z; ce sera une zone aléatoire de la mémoire non volatile.

Selon un deuxième aspect, l'invention prévoit que le verrou est placé à un endroit variable de la mémoire, défini par le système d'exploitation en fonction des espaces disponibles au moment de l'exécution de la section critique. On notera que cette idée est particulièrement inattendue : en effet, la procédure de remise sous tension prévoit de vérifier l'existence d'un verrou, ce qui est évidemment plus difficile si le verrou n'est pas toujours au même endroit, et surtout si on ne sait pas où il est dans la mémoire.

Selon un troisième aspect, l'invention prévoit que le système d'exploitation de la mémoire non volatile fonctionne avec une table d'allocation de blocs de mémoire disponible et avec deux stratégies d'allocation différentes : une stratégie normale et une stratégie d'exception pour certaines informations (dans le cas de la sauvegarde d'informations en vue de l'exécution d'une section critique, ce sont ces informations qui seront stockées selon la stratégie d'exception), la différence entre les deux stratégies étant telle qu'une exploration de la table d'allocation permette de retrouver l'emplacement d'une information qui a été stockée selon la stratégie d'exception.

On voit que sur ce troisième point l'invention s'applique de manière générale à tout système de gestion de mémoire non volatile. En effet, dans les systèmes d'exploitation habituels (surtout pour les mémoires de

masse d'ordinateurs), une seule stratégie d'allocation de blocs disponibles est en général prévue : par exemple une stratégie consistant à rechercher un espace disponible de dimension suffisante et à écrire les informations à partir du premier bloc de cet espace.

La stratégie d'exception pourra consister, selon l'invention, à allouer un espace commençant au deuxième bloc libre suivant un bloc occupé (ce dernier étant de préférence le dernier bloc occupé de la table), en laissant par conséquent un bloc inoccupé entre deux groupes de blocs occupés.

Le repérage de ce bloc inoccupé isolé sera l'indice que l'information stockée dans les blocs suivants a été stockée selon la stratégie d'exception. Ce sera donc en général une indication sur la nature de l'information stockée.

Si on résume les principaux aspects de l'invention, on prévoit d'abord généralement un procédé de stockage d'informations dans une mémoire non volatile, dans lequel l'emplacement physique de mémorisation des informations dans la mémoire est géré par un système d'exploitation permettant d'allouer des emplacements disponibles en fonction du remplissage actuel de la mémoire, à l'aide d'une table d'allocation définissant les blocs de mémoire disponibles et les blocs de mémoire déjà occupés, procédé caractérisé en ce que le système d'exploitation utilise deux stratégies différentes d'allocation d'espace pour les informations à stocker, qui sont respectivement une stratégie normale pour allouer des espaces de mémoire à des informations ordinaires et une stratégie d'exception pour allouer des espaces de mémoire à certaines informations particulières, de manière à permettre de retrouver à partir de la table d'allocation les informations

particulières même si elles sont stockées à un emplacement a priori inconnu.

Et pour l'application plus particulière aux cartes à mémoire, l'invention propose un procédé de fonctionnement d'une carte à mémoire, pour mémoriser une
5 information (INF) dans une mémoire non-volatile de la carte, procédé dans lequel l'emplacement physique de mémorisation des informations dans la mémoire est géré par un système d'exploitation permettant d'allouer des
10 emplacements disponibles en fonction du remplissage actuel de la mémoire, le procédé ayant les particularités suivantes :

- il comporte une écriture en deux phases dans la mémoire, la première phase comprenant une mise en
15 mémoire non volatile d'une information de sauvegarde (I), et la mise en mémoire non volatile d'une information de verrouillage indiquant que la première phase a été exécutée, et la deuxième phase comprenant l'écriture définitive de l'information (INF) puis
20 l'effacement de l'information de verrouillage,

- l'information de sauvegarde (I) est stockée, durant la première phase, à un emplacement variable dans la mémoire non volatile, cet emplacement étant défini par le système d'exploitation en fonction des blocs
25 disponibles dans la mémoire au moment de ce stockage.

L'information de verrouillage est de préférence également stockée à un emplacement variable défini par le système d'exploitation en fonction de l'occupation actuelle de la mémoire.

30 L'information de verrouillage peut alors tout simplement être constituée par l'information de disponibilité des blocs de mémoire contenant l'information de sauvegarde, cette information de disponibilité étant placée dans la table d'allocation

établie par le système d'exploitation.

Dans ce cas, il est tout particulièrement approprié de prévoir une stratégie d'allocation d'exception pour mettre en mémoire l'information de sauvegarde (I), alors
5 qu'une stratégie normale est utilisée pour mettre en mémoire l'information (INF) à mémoriser finalement.

D'autres caractéristiques et avantages de l'invention apparaîtront à la lecture de la description détaillée qui suit et qui est faite en référence aux
10 dessins annexés dans lesquels :

- la figure 1 représente la constitution générale d'une carte à mémoire à microprocesseur;

- la figure 2 représente schématiquement le contenu d'une mémoire non-volatile gérée à partir d'une table
15 d'allocation de fichiers;

- la figure 3 représente un exemple simplifié de contenu de la table d'allocation, et la correspondance avec les emplacements disponibles dans la mémoire;

- la figure 4 représente, pour ce même exemple, les
20 blocs qui sont modifiés pendant la sauvegarde temporaire d'une information critique;

- la figure 5 représente l'écriture d'information définitive pendant une section critique verrouillée;

- la figure 6 représente le déverrouillage après
25 exécution réussie de la section critique d'écriture.

La description détaillée qui va suivre sera faite uniquement à propos d'une carte à mémoire et dans le cas de l'écriture d'une information en deux phases avec
30 sauvegarde d'information avant l'exécution d'une section critique d'un programme d'écriture en mémoire. Cette description sera faite d'abord en supposant que l'information I qu'on veut sauvegarder temporairement en mémoire est justement l'information INF qu'on souhaite

écrire définitivement dans la mémoire. Ceci ne représente qu'un cas particulier de l'invention; c'est le cas le plus facile à décrire et c'est pourquoi il sert de base à la description qui suit.

5 Mais il faut comprendre que des situations plus complexes peuvent se rencontrer, dans lesquelles l'information I dont l'écriture est effectuée pendant une section critique est une information liée à l'information INF à enregistrer mais n'est pas
10 directement cette information à enregistrer.

L'exemple le plus caractéristique de cette situation plus complexe est le cas où les informations à mémoriser sont des fichiers "chaînés". Par fichier chaîné, on entend un fichier dont chaque enregistrement
15 contient l'adresse de l'enregistrement suivant. Dans ce cas, la section critique à verrouiller n'est pas nécessairement la phase d'écriture de l'information, mais ce peut être plutôt la phase de mise à jour de tous les chaînages du fichier. La phase de sauvegarde
20 préalable concernera alors les informations de mise à jour des chaînages.

On va donc d'abord décrire le cas plus simple où une information INF est à écrire dans la carte à mémoire, l'information I sauvegardée temporairement
25 étant essentiellement l'information INF.

Le circuit électronique d'une carte à mémoire à microprocesseur est sommairement représenté à la figure 1. Le circuit comporte un microprocesseur MP auxquels sont reliés un circuit d'interface IT pour la
30 communication avec l'extérieur, une mémoire morte de programmes (ROM) M1, une mémoire vive de travail (RAM) M2, et une mémoire non volatile (EPROM ou EEPROM) M3.

La mémoire de programmes contient notamment le système d'exploitation de la carte, qui gère les

protocoles de communication internes et externes de la carte et qui gère le contenu de la mémoire non volatile M3.

5 Le système d'exploitation de la carte à mémoire utilise une mémoire non volatile pour l'attribution d'emplacements physiques bien déterminés à toute nouvelle information écrite dans la mémoire M3 de la carte. Cette mémoire non volatile est de préférence tout simplement une portion de la mémoire non volatile M3. En
10 général les premiers blocs de la mémoire M3 peuvent servir à cela et ils constituent une table d'allocation de fichiers FAT (ou EAT, Eprom Allocation Table) servant uniquement à la gestion des fichiers de données mémorisées dans la mémoire M3.

15 Sur la figure 2 on a donc représenté la constitution schématique du contenu de la mémoire M3 avec une zone FAT réservée à la gestion de la mémoire. Le reste de la mémoire est une portion MEM dans laquelle peuvent être stockées des informations quelconques.

20 La mémoire MEM (ou plus généralement M3) est divisée en blocs successifs d'informations. Chaque bloc peut comprendre plusieurs mots de mémoire. Les mots peuvent être des octets de huit bits. Les blocs représentent chacun un emplacement physique de mémoire.
25 Les blocs sont de préférence tous de même longueur, par exemple tous de huit octets.

Chaque bloc est représenté par une "image" dans la table d'allocation. Cette image est un bit ou un mot représentant un bloc déterminé. La position d'un mot
30 dans la table d'allocation définit donc de manière biunivoque la position physique d'un bloc de la mémoire MEM.

Le contenu d'un mot dans la table d'allocation peut représenter plusieurs informations, parmi lesquelles

surtout la disponibilité ou l'indisponibilité du bloc de mémoire correspondant. Dans la suite, on considérera que seul un bit de mémoire suffit pour représenter cette disponibilité et donc que la table d'allocation est constituée par une succession de bits, la position d'un bit dans la table représentant de manière biunivoque la désignation d'un bloc de mémoire dans la mémoire MEM, et l'état du bit représentant la disponibilité (bit à zéro par exemple) ou l'indisponibilité (bit à 1) du bloc désigné.

Le système décrit ci-dessus s'inspire simplement des systèmes opératoires de gestion de disques magnétiques dans les ordinateurs.

Sur la figure 2, les bits de la table d'allocation FAT sont désignés par b0, b1, b2, etc., et les blocs correspondants de la mémoire MEM par B0, B1, B2, etc.

Sur la figure 3 on a représenté un exemple très simplifié de contenu de la table d'allocation FAT (ou EAT) avec sa correspondance dans la mémoire MEM : certains blocs sont disponibles, d'autres sont occupés, la table d'allocation est remplie en fonction de cette disponibilité.

Dans l'exemple représenté, on trouve dans la table d'abord des blocs indisponibles (B0,B1), puis peut-être un ou plusieurs blocs libres (B2), de nouveau des blocs occupés (B3,B4,B5), des blocs libres (B6,B7), des blocs occupés (B8,B9,B10,B11), et ainsi de suite jusqu'à ce qu'on ne rencontre plus jusqu'à la fin de la table qu'une succession de blocs libres (ici : B12 et tous les suivants). B11 est ici le dernier bloc occupé de la table.

Pour la gestion normale (sans sauvegarde temporaire et sans protection par un verrou de section critique) du contenu de la mémoire MEM, une étape d'écriture consiste

à demander au système d'exploitation de la mémoire une allocation d'un emplacement de mémoire suffisant pour écrire l'information INF désirée, par exemple P blocs. Le système d'allocation recherche donc dans la table un

5 espace libre de P blocs consécutifs permettant d'écrire l'information INF. La stratégie de recherche et allocation d'espace est toujours la même. Parmi les stratégies les plus connues dans le cas de gestion de disques magnétique, il y a notamment les suivantes :

10 - stratégie dite du "first fit" : le système recherche dans la table d'allocation, en la parcourant depuis le début, le premier espace libre d'au moins P blocs libres consécutifs, et il alloue ces espaces en commençant par le premier bloc de l'espace libre trouvé;

15 - stratégie du "last fit" : recherche du dernier espace libre de longueur suffisante; et allocation de blocs à partir du premier bloc de l'espace trouvé;

20 - stratégie du "best fit" : recherche du plus petit des espaces libres de longueur au moins égale à P blocs, parmi tous les espaces libres de longueur supérieure à P blocs trouvés dans la table; et allocation de blocs à partir du premier bloc libre de l'espace sélectionné.

25 Dans l'exemple qui est donné ici, on suppose par exemple que la stratégie utilisée est la première ("first fit"). Dans l'exemple de la figure 3, l'écriture classique (non protégée) d'une information INF d'au plus deux blocs se ferait dans les blocs B6 et B7, alors que

30 l'écriture d'une information de longueur trois blocs ou plus se ferait à partir du bloc B12.

Selon l'invention, on va utiliser le système d'allocation pour trouver un espace de mémorisation d'une information I temporairement sauvegardée pendant

l'exécution d'une section critique de l'écriture, de sorte que cette information ne sera pas stockée toujours au même endroit.

5 D'autre part on va utiliser comme information de verrouillage les bits de la table d'allocation qui désignent l'emplacement de cette information sauvegardée, de sorte que l'information de verrouillage ne sera pas toujours stockée dans la (ou les) même(s) cellule(s) de mémoire.

10 Enfin, on va utiliser le principe original consistant à utiliser deux stratégies d'allocation différentes selon l'information mise en mémoire : stratégie normale pour allouer un espace de mémorisation des informations habituelles telles que INF, stratégie
15 d'exception pour allouer un espace de mémorisation de l'information sauvegardée I.

C'est l'utilisation de deux stratégies d'allocation différentes qui va alors permettre très simplement de savoir, par simple exploration de la table d'allocation
20 au moment de la remise sous tension de la carte à mémoire, si une information de sauvegarde a été conservée en mémoire; la détection d'une telle information est alors le signe que l'opération d'écriture en section critique s'est déroulée
25 anormalement et qu'il faut reprendre les informations sauvegardées pour les réécrire.

Par conséquent, alors que le système d'exploitation ne sait pas où est le verrou qu'il faut vérifier à la remise sous tension, l'exploration de la configuration
30 de la table lui permettra de trouver à la fois le verrou et l'information sauvegardée, du fait que

1. la configuration de la table laissera apparaître le fait qu'une information a été mémorisée avec la stratégie d'allocation d'exception;

2. la configuration laissera apparaître quels sont les bits de la table d'allocation qui sont concernés et qui constituent le verrou de section critique;

5 3. et enfin la position de ces bits de verrouillage désignera la position (inconnue a priori) de l'information sauvegardée.

De préférence, la stratégie d'exception sera la suivante : on recherche le dernier bloc occupé dans la
10 table (B11 dans l'exemple de la figure 3), on alloue à l'information I à sauvegarder un espace commençant au deuxième bloc libre (B13) après ce dernier bloc occupé. Un bloc libre unique (B12) reste inutilisé.

La procédure d'écriture en deux phases peut se
15 dérouler comme suit :

- première phase : écriture de l'information de sauvegarde I selon la stratégie d'exception : écriture à partir du bloc B13 qui est le deuxième bloc libre après le dernier bloc occupé dans la table;
20 l'information I peut être l'information à mémoriser INF, avec éventuellement d'autres informations, telles qu'un code spécifique indiquant qu'il s'agit bien d'une information de sauvegarde et non d'une information définitive; ce code sert à lever toute ambiguïté sur la
25 détection d'un verrou lors de la remise sous tension de la carte, comme on le verra ultérieurement; les bits b13 et b14 de la table sont mis en même temps à l'état 1 et constituent dès lors le verrou de section critique;

- deuxième phase, qui est la section critique
30 protégée par verrou : écriture définitive de l'information INF selon une stratégie d'allocation normale; si l'information n'utilise que deux blocs elle peut être écrite dans les blocs B6 et B7. Si elle en utilise plus elle est écrite en commençant au bloc B15.

La figure 5 représente la table d'allocation FAT et la mémoire MEM à ce stade.

Si tout s'est passé correctement, l'étape finale de l'écriture consiste à effacer le verrou, marquant ainsi
5 la fin de la section critique.

Cet effacement consiste simplement à libérer les blocs B13 et B14 contenant l'information sauvegardée, en remettant à zéro les bits correspondants b13 et b14 de la table d'allocation.

10 La figure 6 représente la mémoire à ce stade. Elle est prête pour d'autres opérations.

Si l'opération d'écriture a été interrompue anormalement pendant la section critique, la configuration au moment de l'interruption est celle de
15 la figure 4 ou la figure 5 : verrou positionné.

La configuration de la table d'allocation est alors particulière en ce sens que le dernier bit à zéro (b12) qui précède le dernier groupe de bits à 1 (b13, b14 figure 4 ou b13 à b19, figure 5) est un bit unique à
20 zéro entre deux groupes de bits à 1. Autrement dit il y a un bloc libre B12 isolé entre des blocs occupés, et ce bloc libre est le dernier bloc libre dans la partie effectivement utilisée de la mémoire.

La procédure de remise sous tension de la carte à
25 mémoire comportera obligatoirement une procédure de détection d'une telle configuration. Cette configuration résulte en effet a priori de ce que certaines informations ont été écrites avec une stratégie d'exception différente de la stratégie normale
30 d'écriture des autres informations. Cette procédure de détection comportera une recherche du dernier groupe de bits à 1 dans la table d'allocation, et une recherche du dernier bit à zéro précédant ce groupe de bits à 1. Si ce bit à zéro est encadré de bits à 1 il y a une

probabilité forte pour qu'il soit le signe de la présence d'un verrou.

Pour effectuer cette recherche, la table d'allocation sera de préférence parcourue en sens inverse du sens normal, c'est-à-dire en remontant depuis la fin de la table, jusqu'à ce qu'on rencontre le premier groupe de bits à 1 puis le premier bit à 0 et enfin en examinant le bit caractéristique suivant au cours de cette remontée :

- s'il est à 0 il n'y a pas de verrou positionné; c'est le cas pour la figure 6 : le bit b14, premier bit à zéro en partant de la fin n'est pas encadré par deux bits à 1.

- si au contraire il est à 1, il y a très probablement un verrou positionné; c'est le cas pour la configuration de la figure 5 ou 4 : le bit b12, premier bit à zéro en partant de la fin est encadré par deux bits à 1.

Dans ce dernier cas, la procédure de mise en route de la carte, effectuée par le système d'exploitation, comporte la reprise de la deuxième étape d'écriture; cette reprise est protégée par le verrou de section critique (b13, b14) qui n'a pas changé d'état. Le système d'allocation attribue (selon la stratégie normale d'allocation) des blocs disponibles pour l'information INF. Et l'information INF, recueillie dans les emplacements désignés par le verrou, est écrite dans ces emplacements. Après cela, le verrou est effacé.

Il peut y avoir une ambiguïté sur la configuration de mémoire qui fait apparaître un verrou. Il se peut en effet que dans certains cas, la stratégie d'écriture normale ("first fit", "last fit", "best fit" ou autre) aboutisse à une configuration avec un bloc libre isolé entre deux blocs occupés, ce bloc libre étant le dernier

bloc libre avant la fin de la zone occupée de la table d'allocation. Dans ce cas, la configuration laisse croire qu'un verrou a été placé en utilisant la stratégie d'exception, alors que ce n'est pas vrai.

5 Pour éliminer toute ambiguïté dans ce cas exceptionnel, on prévoit que les blocs alloués pour l'information de sauvegarde I mémorisent non seulement l'information qu'on désire sauvegarder, mais aussi un code indiquant qu'il s'agit bien d'une information de
10 sauvegarde. Si ce code n'est pas trouvé à la lecture des blocs désignés par le verrou, la reprise de l'écriture n'est pas effectuée.

Pour éviter une fragmentation excessive de la mémoire M3 c'est à dire pour éviter d'avoir trop de
15 blocs libres encadrés par des blocs occupés, le système d'exploitation peut tout d'abord réserver l'espace définitif nécessaire à l'information INF. Dans ce cas on aura pour reprendre l'illustration de la figure 5, b12 à b15 alloués à l'information définitive "1111".

20 Ensuite lorsque l'espace de sauvegarde I est alloué on aura b12 à b15 = 1 1 1 1 (INF)

b16 à b19 = 0 1 1 1 1 (I)

Ainsi la libération de l'espace alloué pour la sauvegarde I, ne générera pas de "trou" une fois
25 l'écriture terminée :

b8 à b15 = 1 1 1 1 1 1 1 1

b16 à b23 = 0 0 0 0 0 0 0 0

L'invention s'applique également lorsque l'information sauvegardée I n'est pas l'information INF
30 à écrire définitivement dans la mémoire.

Un exemple simple peut en être donné dans le cas où la mémoire contient des fichiers sous forme d'une suite d'enregistrements chaînés. L'enregistrement de rang i du fichier contient l'adresse de l'enregistrement de rang

i+1, ce qui permet de parcourir dans un sens la chaîne. Si le fichier doit pouvoir être parcouru dans les deux sens, il faut que l'enregistrement de rang i contienne aussi l'adresse de l'enregistrement de rang i-1.

5 L'ajout d'un nouvel enregistrement dans le fichier peut se faire de la manière suivante :

 - première phase : demande au système d'exploitation d'un espace de mémoire suffisant pour l'écriture de l'enregistrement; positionnement des bits
10 correspondants dans la table d'allocation selon une stratégie normale d'allocation; puis écriture des informations à l'emplacement alloué (cet enregistrement, bien que définitif, ne constitue pas une section critique de la séquence d'écriture, contrairement au cas
15 décrit précédemment); puis demande d'un espace libre pour une information à sauvegarder en vue d'une section critique (l'information à sauvegarder est l'information nécessaire à la mise à jour des chaînages de fichiers); allocation de cet espace selon la stratégie d'exception;
20 écriture dans cet espace des informations nécessaires à la mise à jour des chaînages; et enfin positionnement à 1 des bits de la table d'allocation qui désignent l'emplacement des informations sauvegardées et qui vont servir de verrou;

25 - deuxième phase, qui est la section critique, qui devra être exécutée à nouveau en cas d'arrachement de la carte au cours de cette phase : mise à jour définitive du chaînage;

30 Les bits de verrouillage sont remis ensuite à zéro si tout s'est bien passé.

 Dans ce cas, pour distinguer un verrou d'une information ordinaire lorsque la configuration de la table d'allocation ressemble par hasard à une configuration avec verrou, on utilise le fait que les

enregistrements comportent tous une adresse d'enregistrement suivant ou précédent. Les blocs contenant les informations sauvegardées pourront contenir une adresse impossible (adresse hors de la
5 mémoire M3) à la place d'une adresse d'enregistrement suivant. Ceci permettra de vérifier que les blocs désignés par les bits de verrouillage sont bien des blocs de sauvegarde.

Un code de redondance utilisant la longueur de
10 l'enregistrement de sauvegarde peut être également placé en fin de l'enregistrement de sauvegarde, pour assurer que des informations n'ont pas été perdues pendant l'écriture en mémoire non volatile des informations de sauvegarde.

15 On a ainsi décrit un cas dans lequel l'information sauvegardée I n'est pas exactement l'information INF à mémoriser définitivement.

Bien entendu la stratégie d'exception peut être différente. L'allocation de l'espace peut commencer au
20 3ème, 4ème, 5ème bloc libre suivant le dernier bloc occupé.

REVENDEICATIONS

1. Procédé de stockage d'informations dans une mémoire non volatile, dans lequel l'emplacement physique de mémorisation des informations dans la mémoire est géré par un système d'exploitation permettant d'allouer des emplacements disponibles en fonction du remplissage actuel de la mémoire, à l'aide d'une table d'allocation (FAT) définissant les blocs de mémoire disponibles et les blocs de mémoire déjà occupés, procédé caractérisé en ce que le système d'exploitation utilise deux stratégies différentes d'allocation d'espace pour les informations à stocker, qui sont respectivement une stratégie normale pour allouer des espaces de mémoire à des informations ordinaires (INF) et une stratégie d'exception pour allouer des espaces de mémoire à certaines informations particulières (I), de manière à permettre de retrouver à partir de la table d'allocation les informations particulières même si elles sont stockées à un emplacement inconnu.

2. Procédé selon la revendication 1, caractérisé en ce que la stratégie d'allocation d'exception consiste à allouer un espace (b13, b14) commençant au deuxième bloc libre suivant un bloc occupé (b11).

3. Procédé selon la revendication 2, caractérisé en ce que la stratégie d'exception consiste à allouer un espace commençant au deuxième bloc libre suivant le dernier bloc occupé de la table (b11).

4. Procédé selon l'une des revendications 1 à 3, caractérisé en ce que la stratégie normale consiste à allouer un espace commençant au premier bloc libre suivant un bloc occupé et faisant partie d'un espace

libre de dimension suffisante.

5. Procédé de fonctionnement d'une carte à mémoire, pour mémoriser une information (INF) dans une mémoire non-volatile (MEM) de la carte, procédé dans lequel l'emplacement physique de mémorisation des informations dans la mémoire est géré par un système d'exploitation permettant d'allouer des emplacements disponibles en fonction du remplissage actuel de la mémoire, le procédé ayant les particularités suivantes :

10 - il comporte une écriture en deux phases dans la mémoire, la première phase comprenant une mise en mémoire non volatile d'une information de sauvegarde (I), et la mise en mémoire non volatile d'une information de verrouillage (b13, b14) indiquant que la première phase a été exécutée, et la deuxième phase comprenant l'écriture définitive de l'information (INF) puis l'effacement de l'information de verrouillage,

15 - l'information de sauvegarde (I) est stockée, durant la première phase, à un emplacement variable dans la mémoire non volatile, cet emplacement étant défini par le système d'exploitation en fonction des blocs disponibles dans la mémoire au moment de ce stockage.

6. Procédé selon la revendication 1, caractérisé en ce que l'information de verrouillage est stockée à un emplacement variable défini par le système d'exploitation.

7. Procédé selon la revendication 6, caractérisé en ce que le système d'exploitation utilise une table d'allocation définissant les blocs de mémoire disponibles, l'information de verrouillage étant constituée par les bits de la table d'allocation qui désignent les blocs de mémoire occupés par l'information de sauvegarde.

8. Procédé selon la revendication 7, caractérisé

en ce que le système d'exploitation utilise deux stratégies différentes d'allocation qui sont une stratégie normale pour mémoriser l'information à écrire dans la mémoire et une stratégie d'exception pour
5 mémoriser l'information de sauvegarde, l'utilisation de la stratégie d'exception créant une configuration de la table d'allocation telle qu'on puisse détecter dans la table la présence de l'information de verrouillage placée à l'aide de la stratégie d'exception même si
10 l'emplacement de cette information de verrouillage est a priori inconnue.

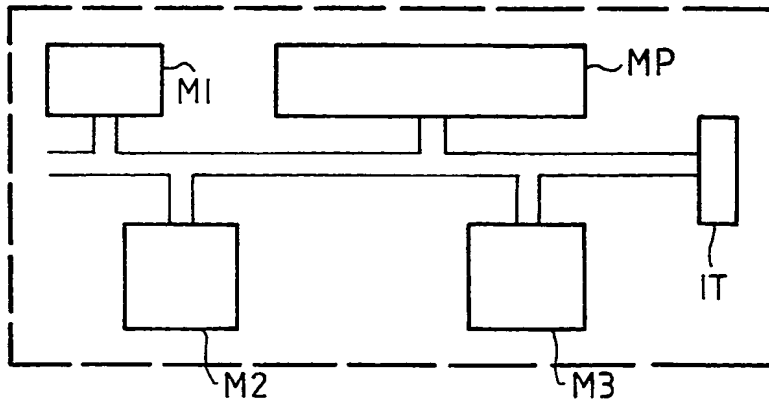
9. Procédé selon la revendication 8, caractérisé en ce que la stratégie d'allocation d'exception consiste à allouer un espace commençant au deuxième bloc libre
15 suivant un bloc occupé.

10. Procédé selon la revendication 9, caractérisé en ce que la stratégie d'exception consiste à allouer un espace commençant au deuxième bloc libre suivant le dernier bloc occupé de la table.

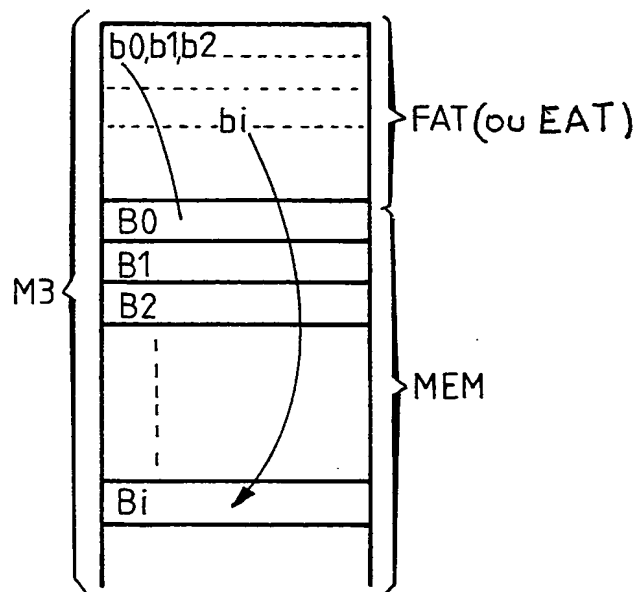
20 11. Procédé selon l'une des revendications 8 à 10, caractérisé en ce que la stratégie normale consiste à allouer un espace commençant au premier bloc libre suivant un bloc occupé et faisant partie d'un espace libre de dimension suffisante.

25 12. Procédé selon l'une des revendications 8 à 11, caractérisé en ce que, lors de la mise sous tension de la carte à mémoire, le système d'exploitation recherche dans la table d'allocation le dernier groupe de bits désignant des blocs occupés, puis le bit désignant le
30 bloc libre précédant ce groupe, et vérifie l'état du bit caractéristique précédant ce bit désignant un bloc libre, et en ce qu'une opération de reprise d'écriture est effectuée si le bit caractéristique correspond à un bloc occupé.

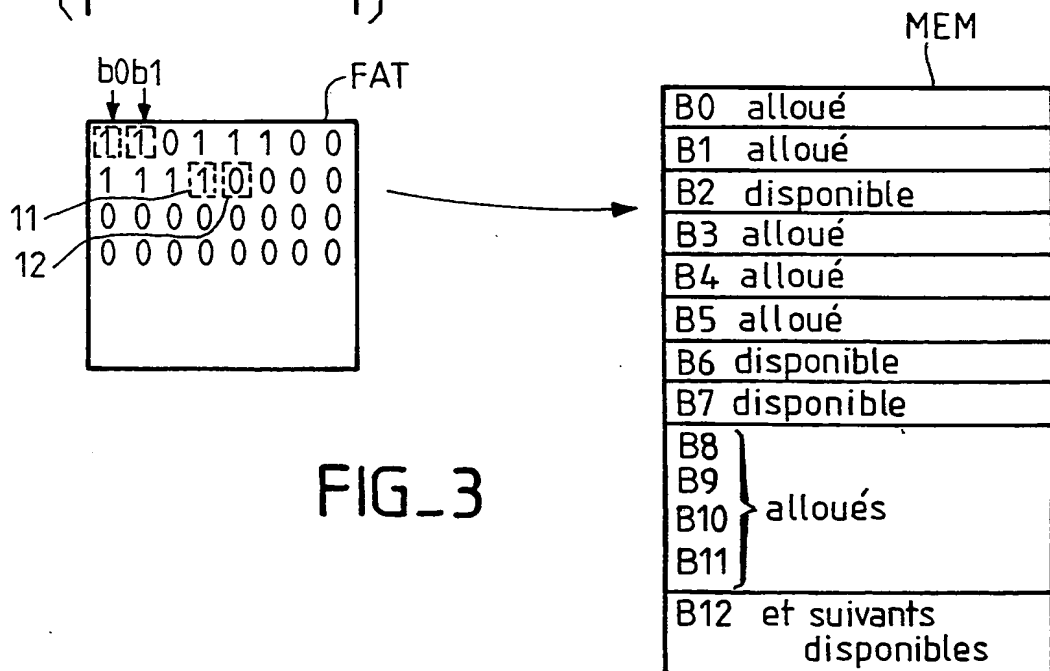
1/2



FIG_1



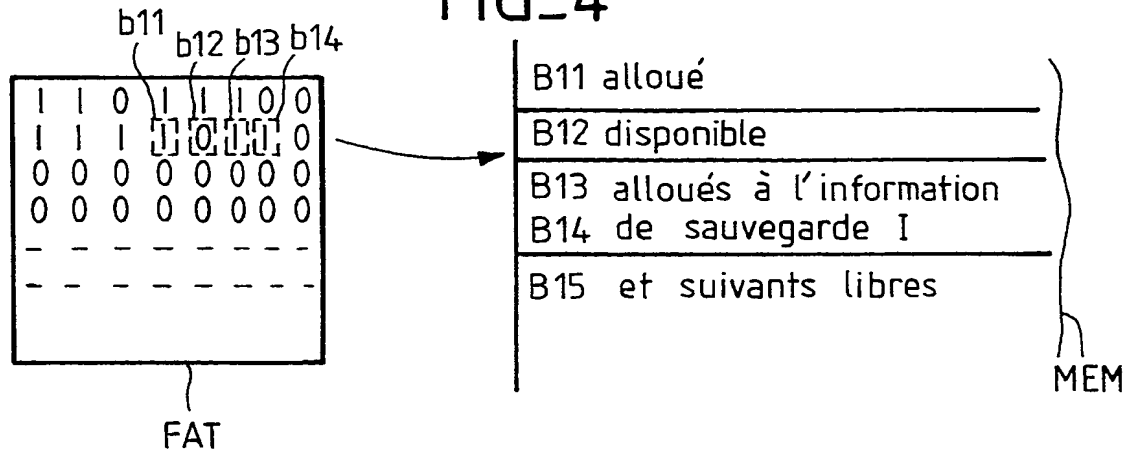
FIG_2



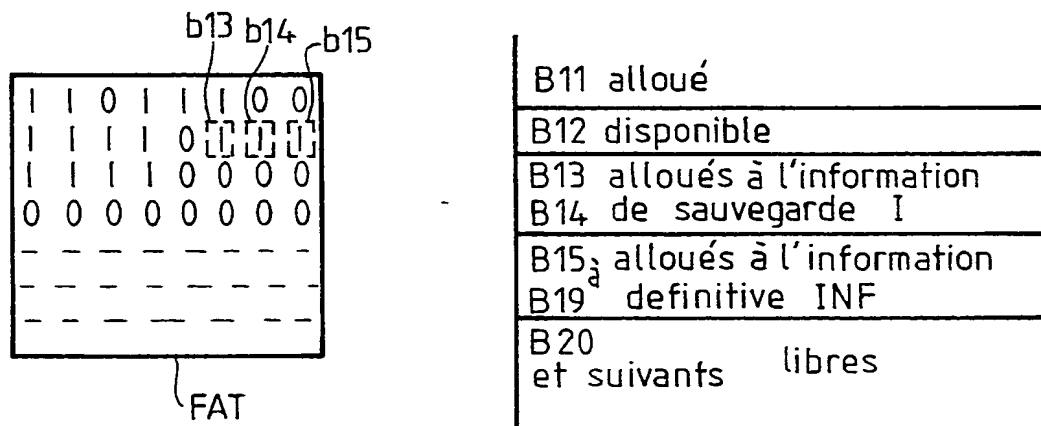
FIG_3

2/2

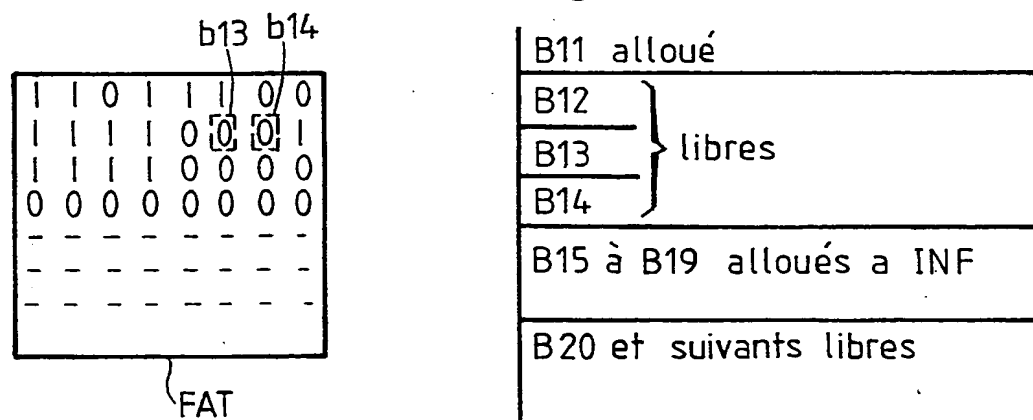
FIG_4



FIG_5



FIG_6



INSTITUT NATIONAL
de la
PROPRIETE INDUSTRIELLERAPPORT DE RECHERCHE
établi sur la base des dernières revendications
déposées avant le commencement de la rechercheFR 9301735
FA 483328

DOCUMENTS CONSIDERES COMME PERTINENTS		Revendications concernées de la demande examinée
Catégorie	Citation du document avec indication, en cas de besoin, des parties pertinentes	
X	US-A-4 277 826 (COLLINS ET AL) * abrégé; figure 1 * * colonne 3, ligne 14 - colonne 5, ligne 13 *	5-8
Y	---	9-12
Y	EP-A-0 400 475 (FUJI PHOTO FILM CO.) * abrégé; figures 2,7-9 * * colonne 8, ligne 17 - colonne 11, ligne 36 *	1-4, 12
Y	--- IBM TECHNICAL DISCLOSURE BULLETIN, vol. 22, no. 6, Novembre 1979, NEW YORK, US; pages 2317 - 2318 P.G.CASPERS ET AL 'Cache-Resident Processor Registers' * le document en entier *	1-4
Y	US-A-5 142 676 (FRIED ET AL) * abrégé; figure 1 * * revendications 1-4 *	2-4, 9-11
		DOMAINES TECHNIQUES RECHERCHES (Int. Cl.5)
		G06F G07F
Date d'achèvement de la recherche 18 AOUT 1993		Examinateur POWELL D.
CATEGORIE DES DOCUMENTS CITES X : particulièrement pertinent à lui seul Y : particulièrement pertinent en combinaison avec un autre document de la même catégorie A : pertinent à l'encontre d'au moins une revendication ou arrière-plan technologique général O : divulgation non-écrite P : document intercalaire T : théorie ou principe à la base de l'invention E : document de brevet bénéficiant d'une date antérieure à la date de dépôt et qui n'a été publié qu'à cette date de dépôt ou qu'à une date postérieure. D : cité dans la demande L : cité pour d'autres raisons & : membre de la même famille, document correspondant		

EPO FORM 1503 01.82 (P0412)

This Page Blank (uspto)